**Flight Price Prediction Using Machine Learning**

-By Avinash Patel

In this article, I will be importing a dataset with several columns and rows and try to build an ML model to predict the price of a flight based on the data of independent columns.

As we all know the advantages of Machine Learning are vast. It helps us to create ways of **modernizing** technology. One of its applications is predicting the outcome based on the previously known attributes on which the model is trained. By predicting the outcome we can find an estimate on how much the cost will be for better planning of budget and planning for trips accordingly. Many factors are affecting the price of flight all of them are represented in the form of columns in the dataset which we will see further in the article.

Problem Statement

Flight ticket prices can be something hard to guess. Today we might see some price of a flight ticket and tomorrow the same ticket might be different in price according to some attributes which have changed due to the change in time and date.

To solve this problem, we have been provided with prices of flight tickets for various airlines with various input features and we will build an ML model based on these input features (Dependent Variables) to predict our output feature (Target Variable).

# The Dataset

Our dataset contains 10683 rows and 11 columns.

**Following is the description of features available in the dataset :**

1. **Airline**: The name of the airline.

2. **Date\_of\_Journey**: The date of the journey

3. **Source**: The source from which the service begins.

4. **Destination**: The destination where the service ends.

5. **Route**: The route taken by the flight to reach the destination.

6. **Dep\_Time**: The time when the journey starts from the source.

7. **Arrival\_Time**: Time of arrival at the destination.

8. **Duration**: Total duration of the flight.

9. **Total\_Stops**: Total stops between the source and destination.

10. **Additional\_Info**: Additional information about the flight

11. **Price**: The price of the ticket

**Unique variables in our columns are:**

Airline 12

Date\_of\_Journey 44

Source 5

Destination 6

Route 128

Dep\_Time 222

Arrival\_Time 1343

Duration 368

Total\_Stops 5

Additional\_Info 10

Price 1870

**Now checking how amny null values are there:**

Route 1

Total\_Stops 1

As only two rows have null values so we will delete those rows.

# Exploratory Data Analysis

# Now, lets look at our dataset and find out the simple details like what is the maximums and the minimums and other things like the standard deviation and mean:

# 

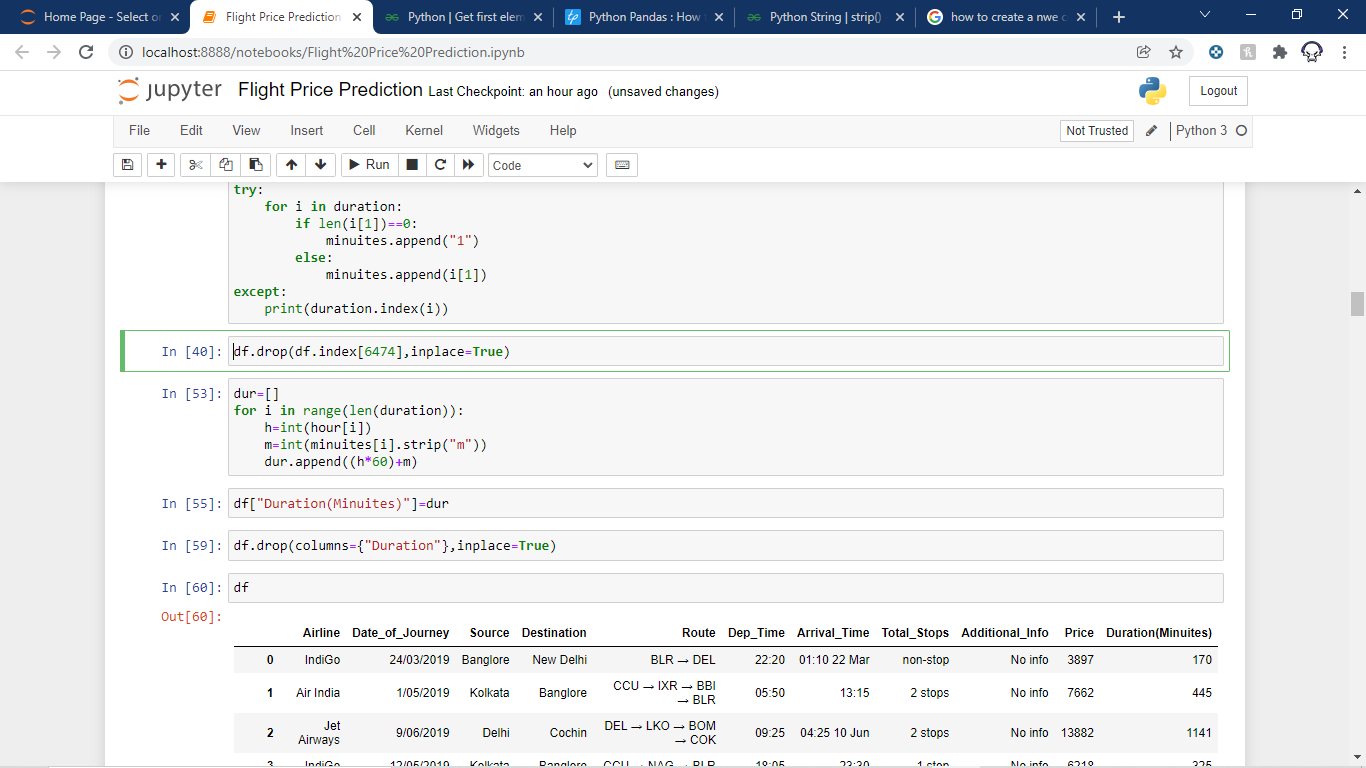
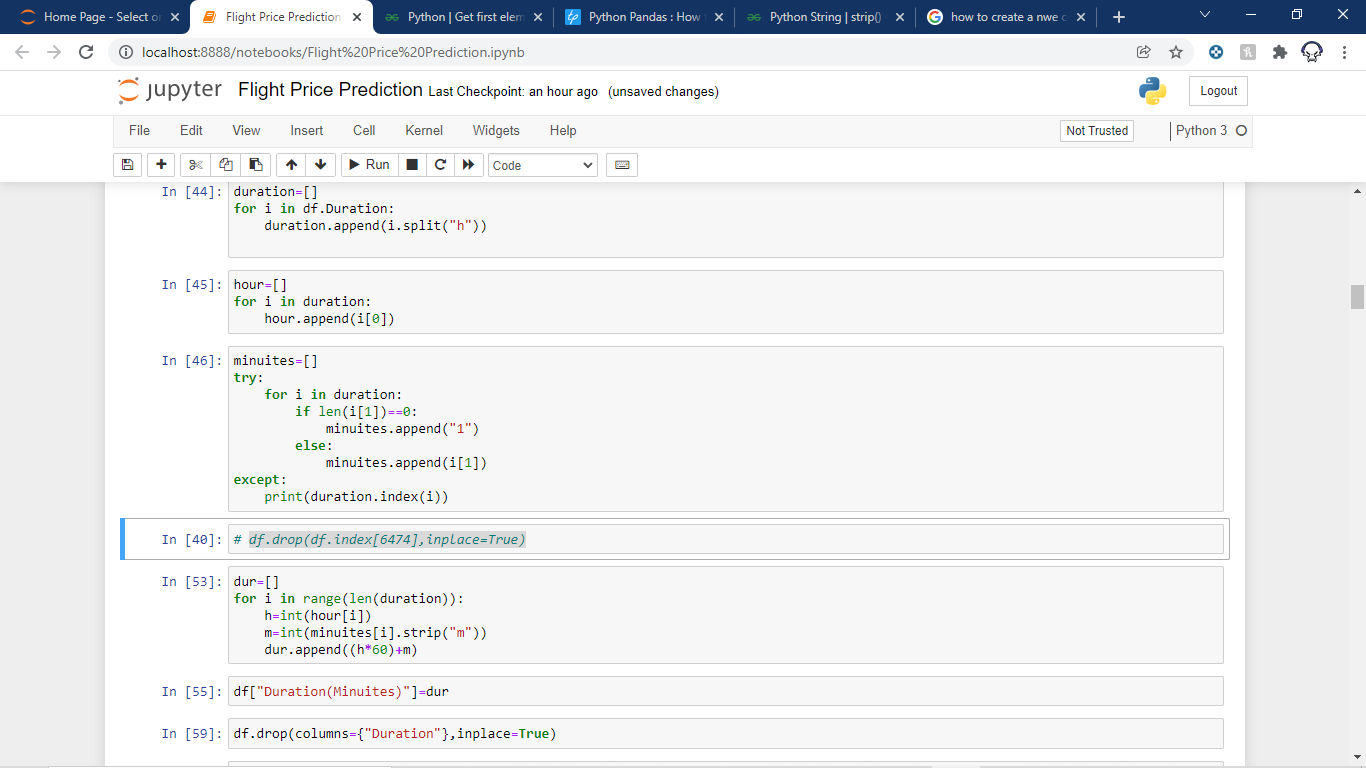
We identify the below mentioned points in the first look:

1: Our price of the flight could be as high as 79512 but the average price is 9087.

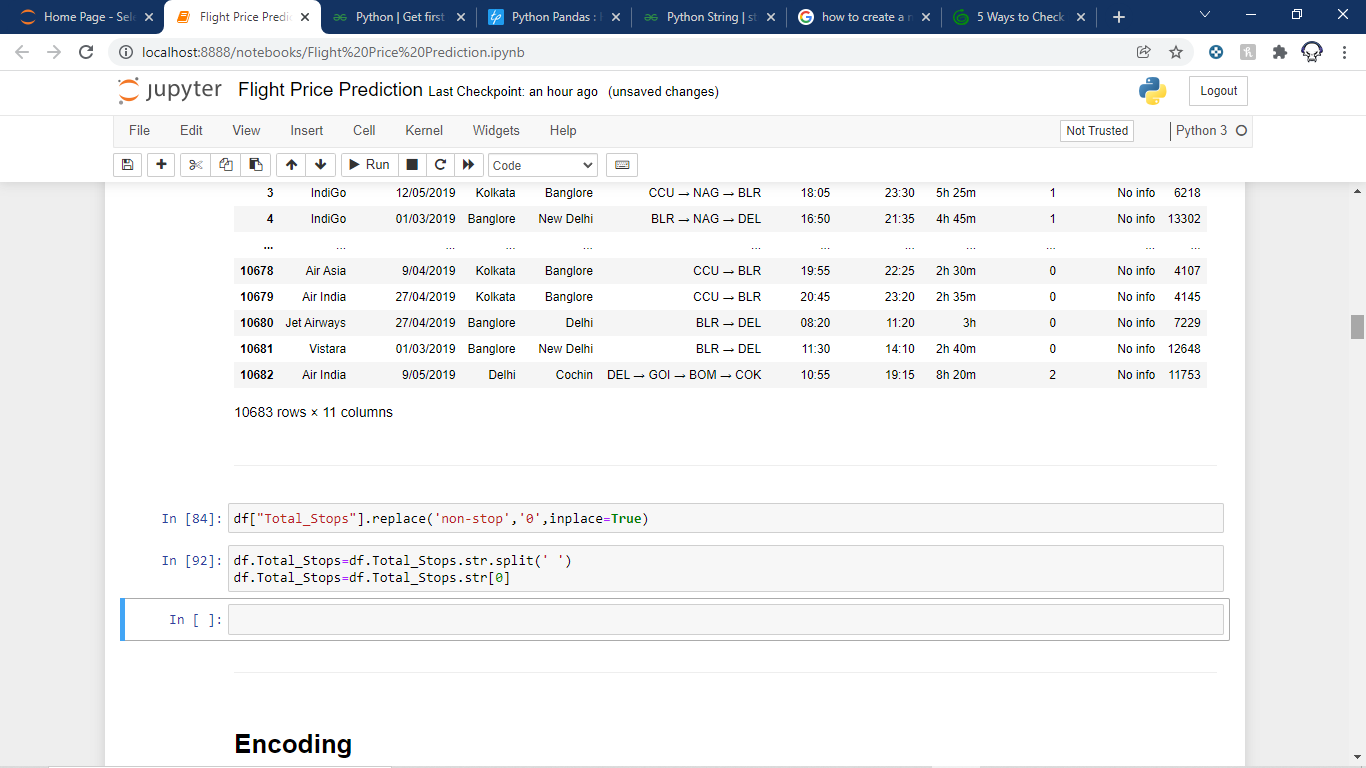
2: The average duration of a flight is 2h 50min .

Now we will convert the “Duration” column to “Duration(Minutes)” column.

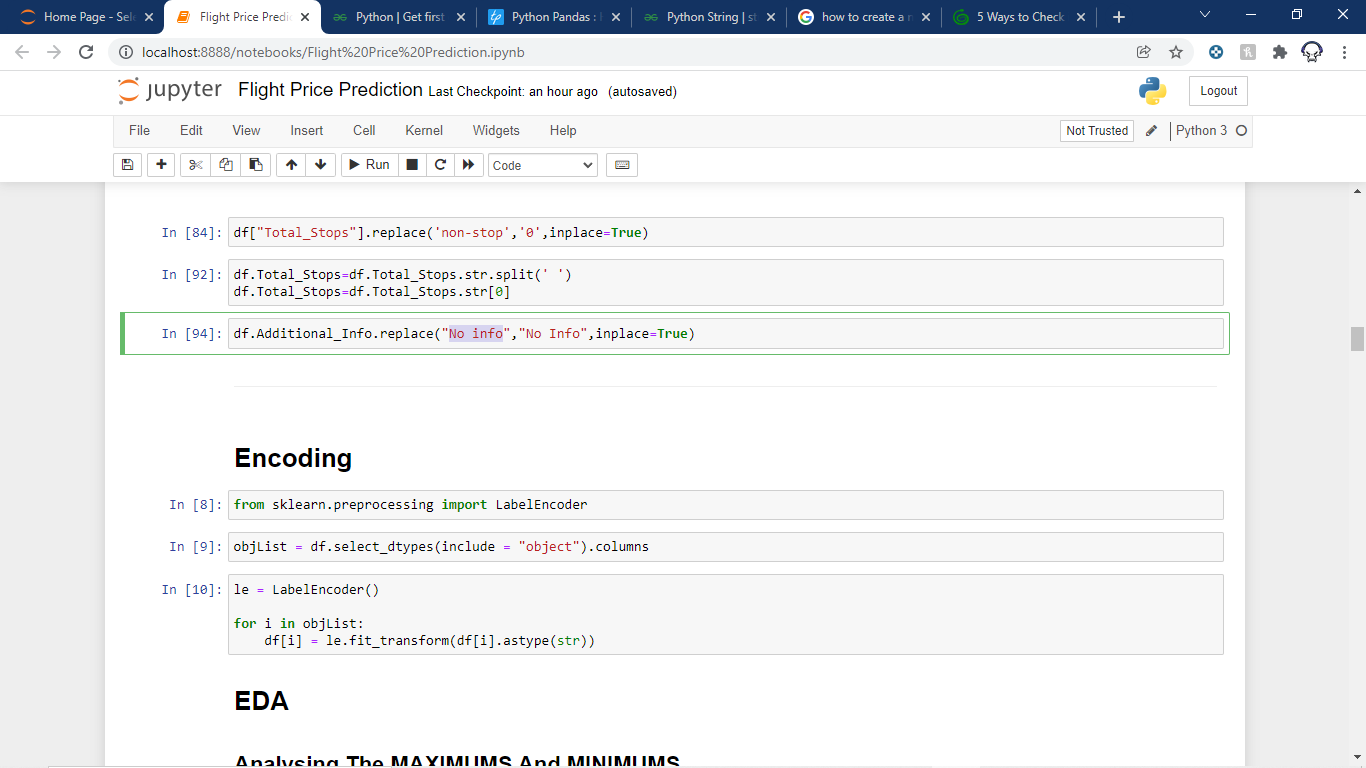
While Doing so I encountered an unexpected error while separating the minutes from the Duration so I put up a try catch block to see in which column does the error is occurring and I find out that in index 6474 the duration is just 5 minutes which clearly an error so I dropped the row. Then I finally calculated the minutes and dropped the Duration column.



Now we will convert the “Total\_stops” column as we will set all the non-stop blocks as 0 and then we will split the number of stops and the “stops” word from the column and place only the numbers in the column values.

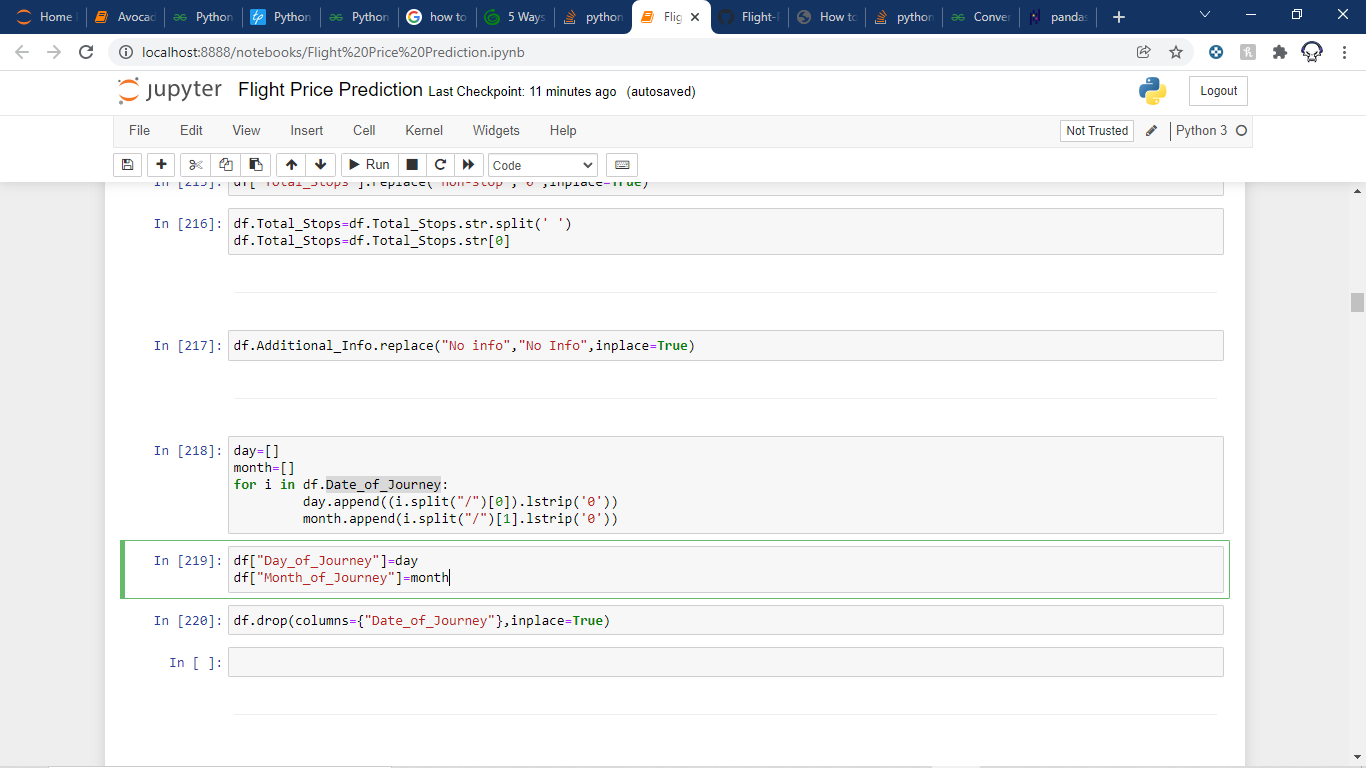


In “Additional\_Info” column ther are two values “No info”,” No Info” which is treated as different values so we will replace one of them with the other.



By doing so I calculated that the percentage of No Info is more than 80% so we drop this column.

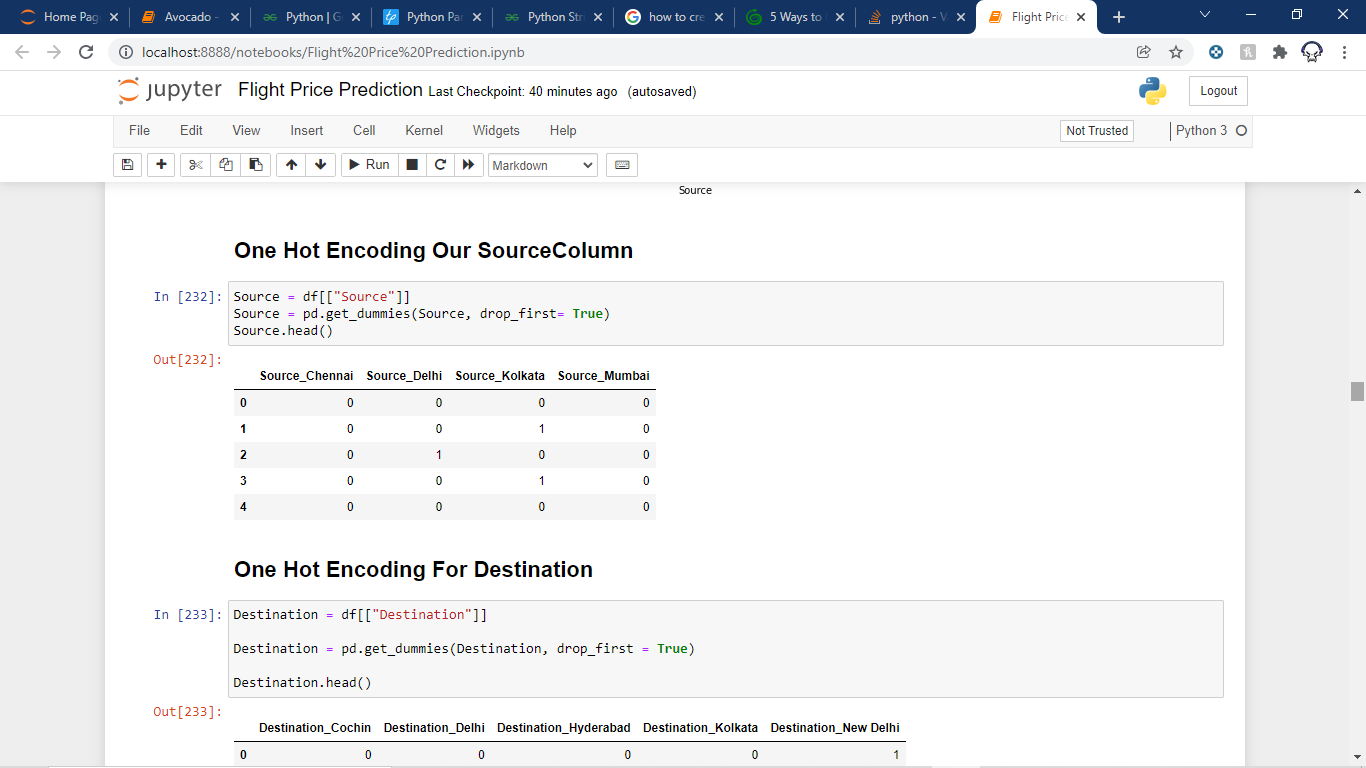
Now we will split “Date\_of\_Journey” into “Day\_of\_Journey” and “Month\_of\_Journey” as the year is irrelevant for us and drop that column.

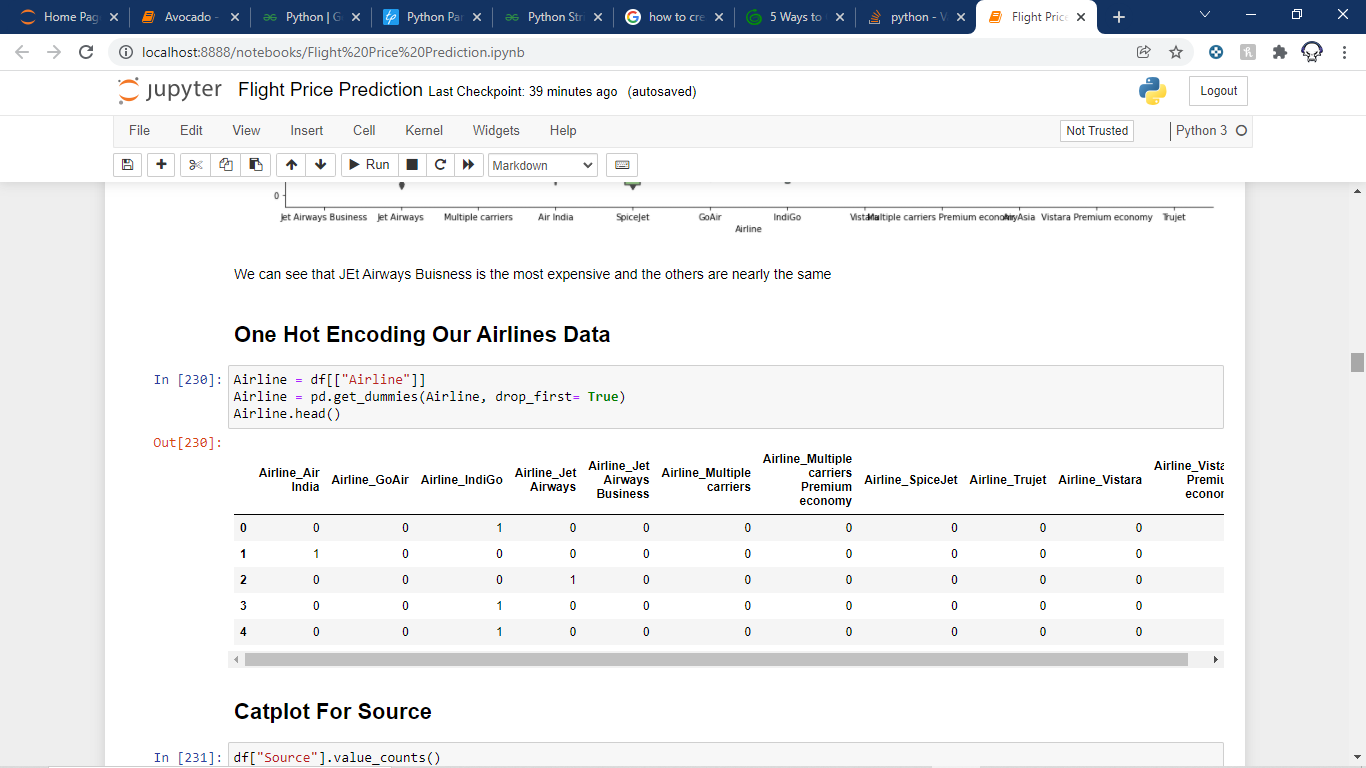


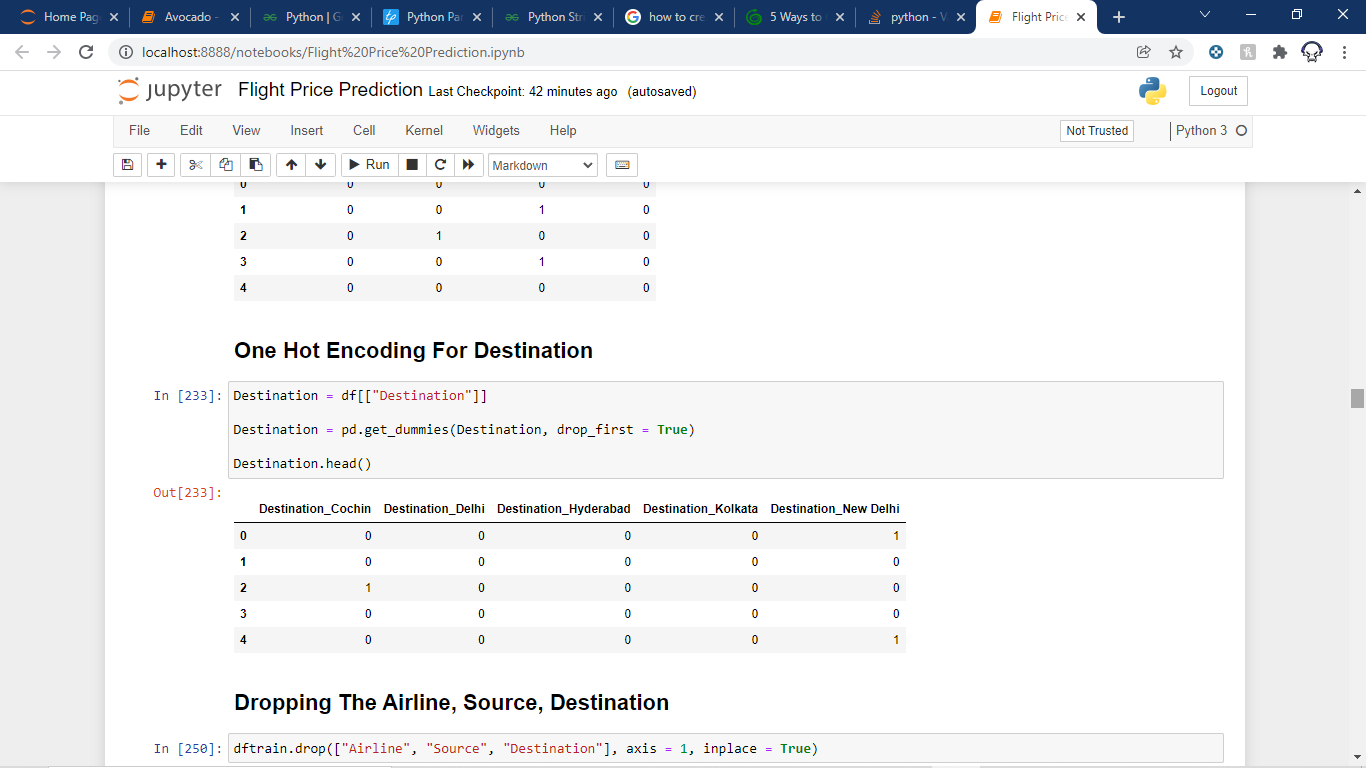
## Now we Extract hour and minute from “Arrival time” and “Departure time” and delete these columns as well.

## 

As our Airlines, Destination and Source are Categorical columns and all of them has nominal data so we will do ONE-HOT Encoding on them and delete those columns.



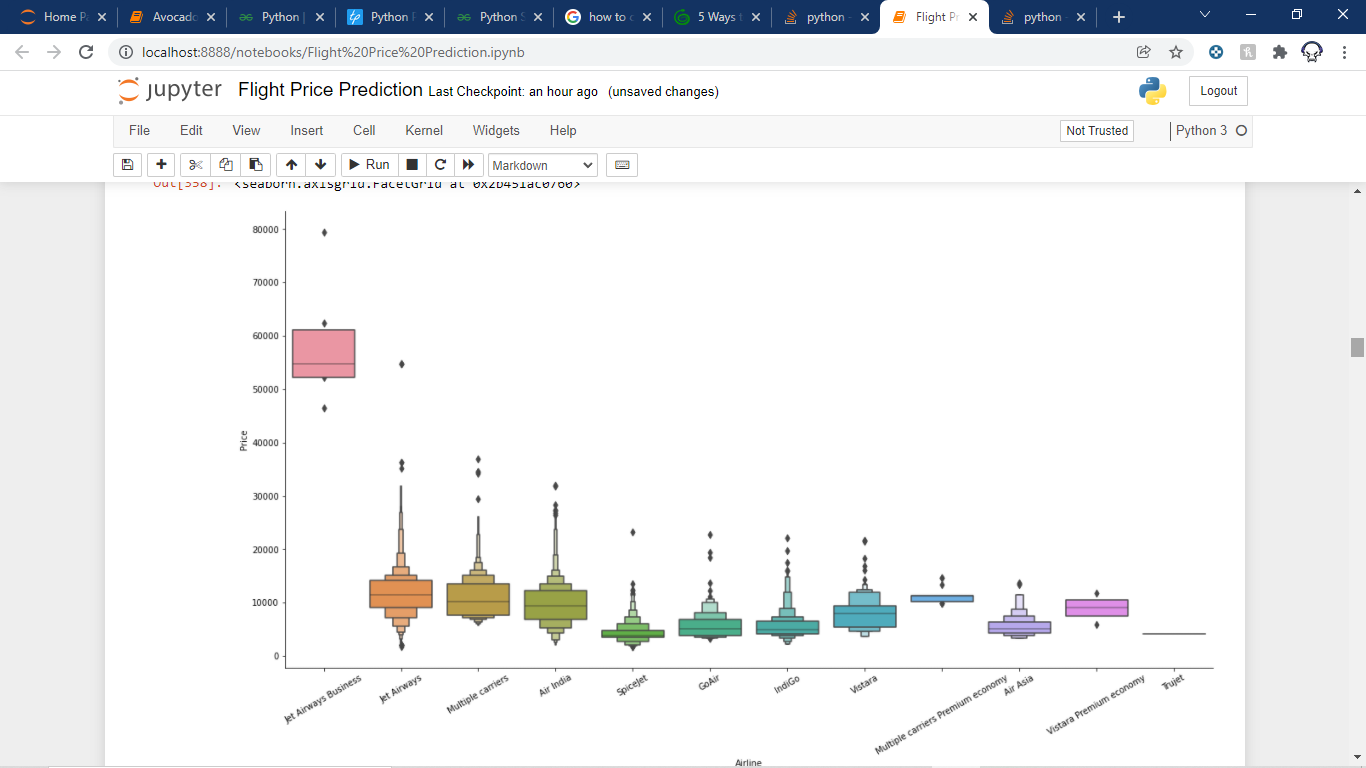




Now we are left with a dataset that has all numeric columns.

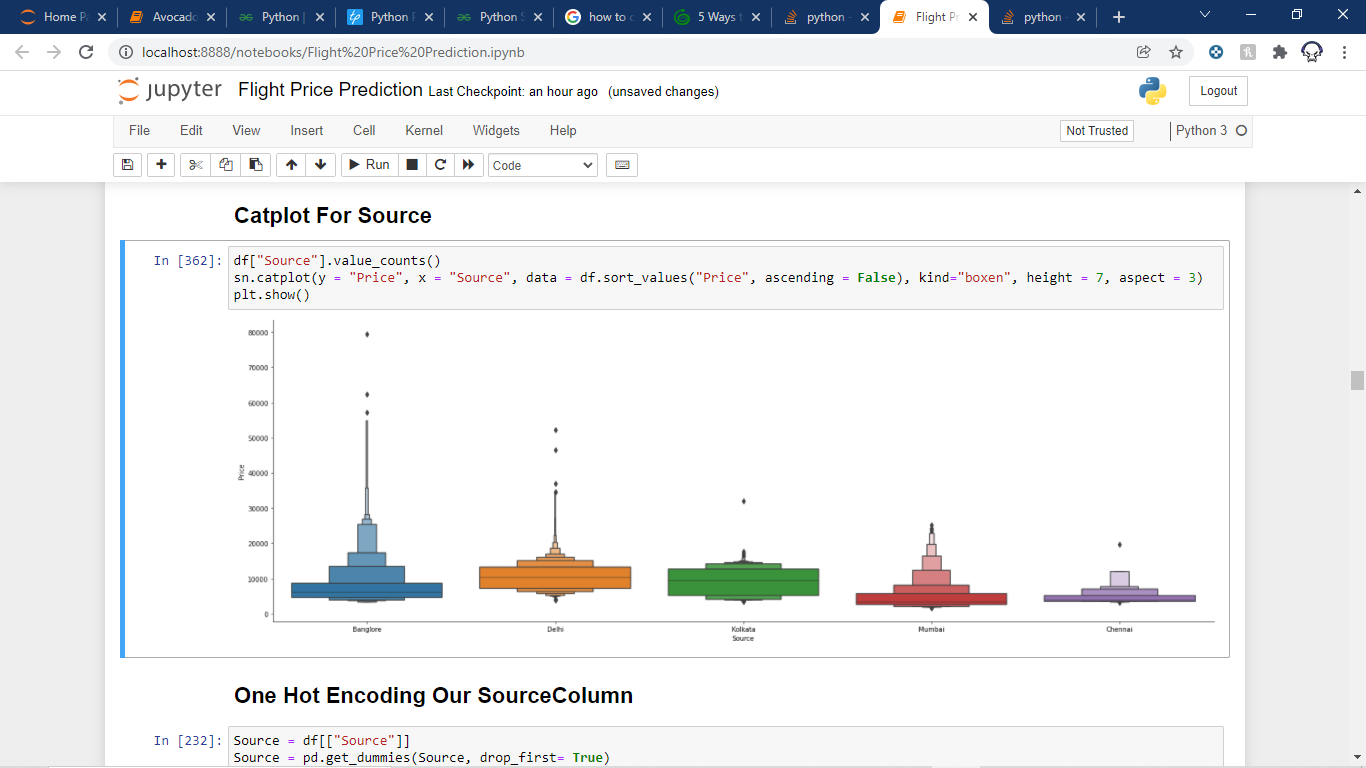
Now let us analyse some columns:

## Catplot for Airline:



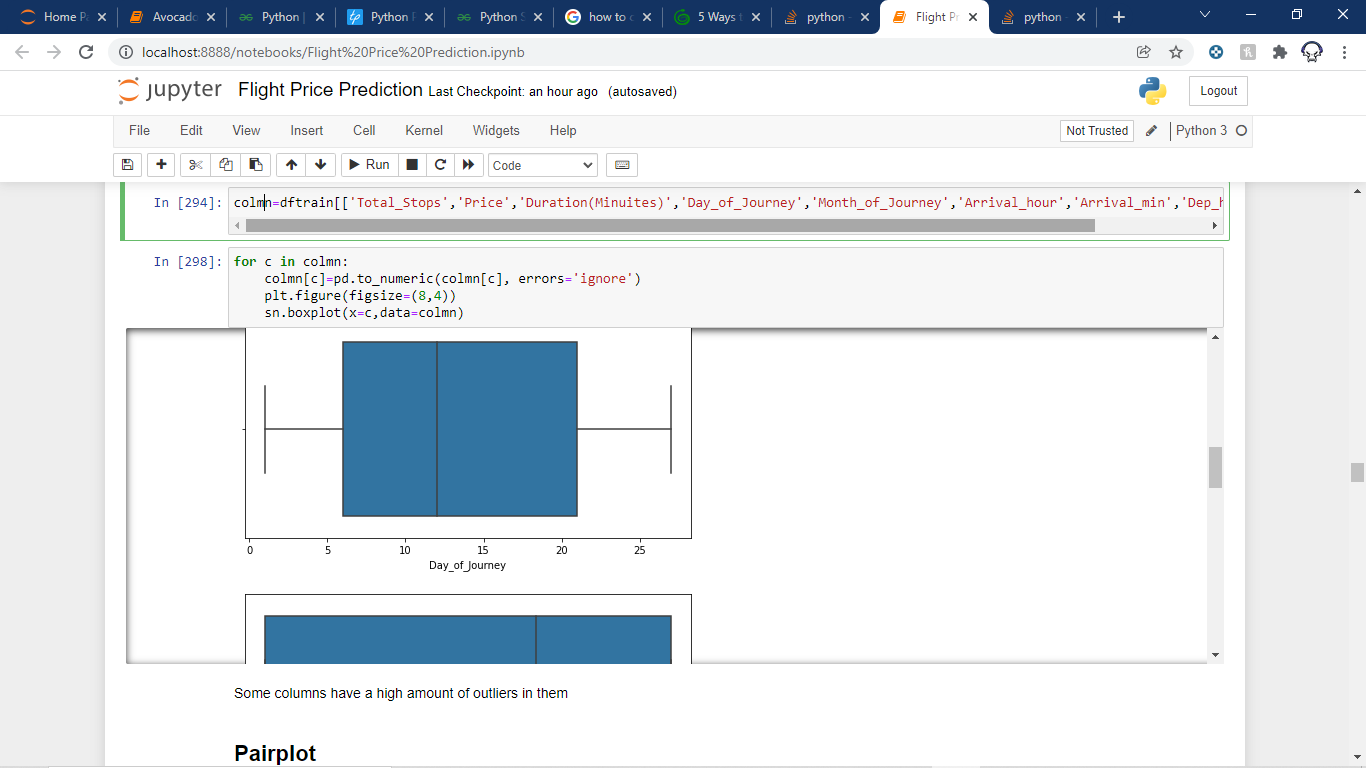
We can see that Jet Airways Business is the most expensive and kind of an outlier and the others are nearly the same price.

Catplot for Source:



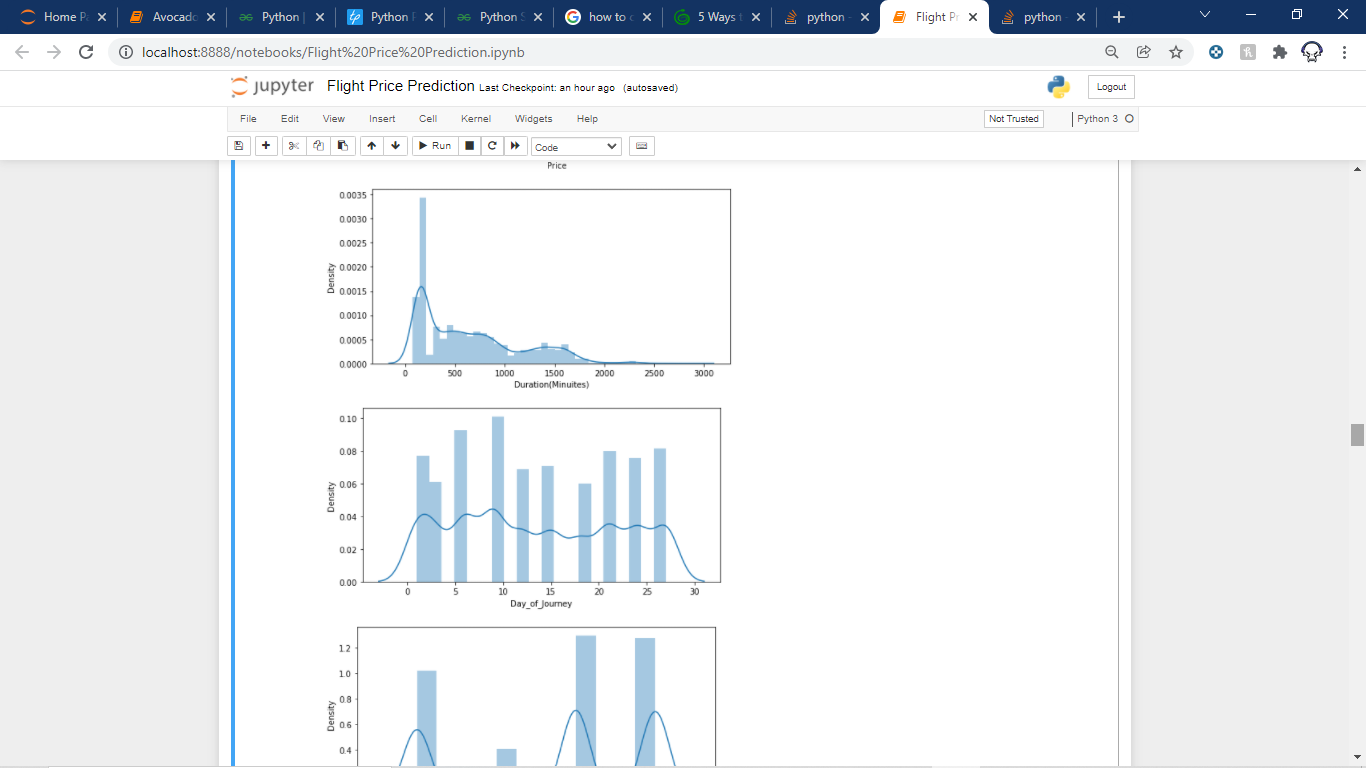
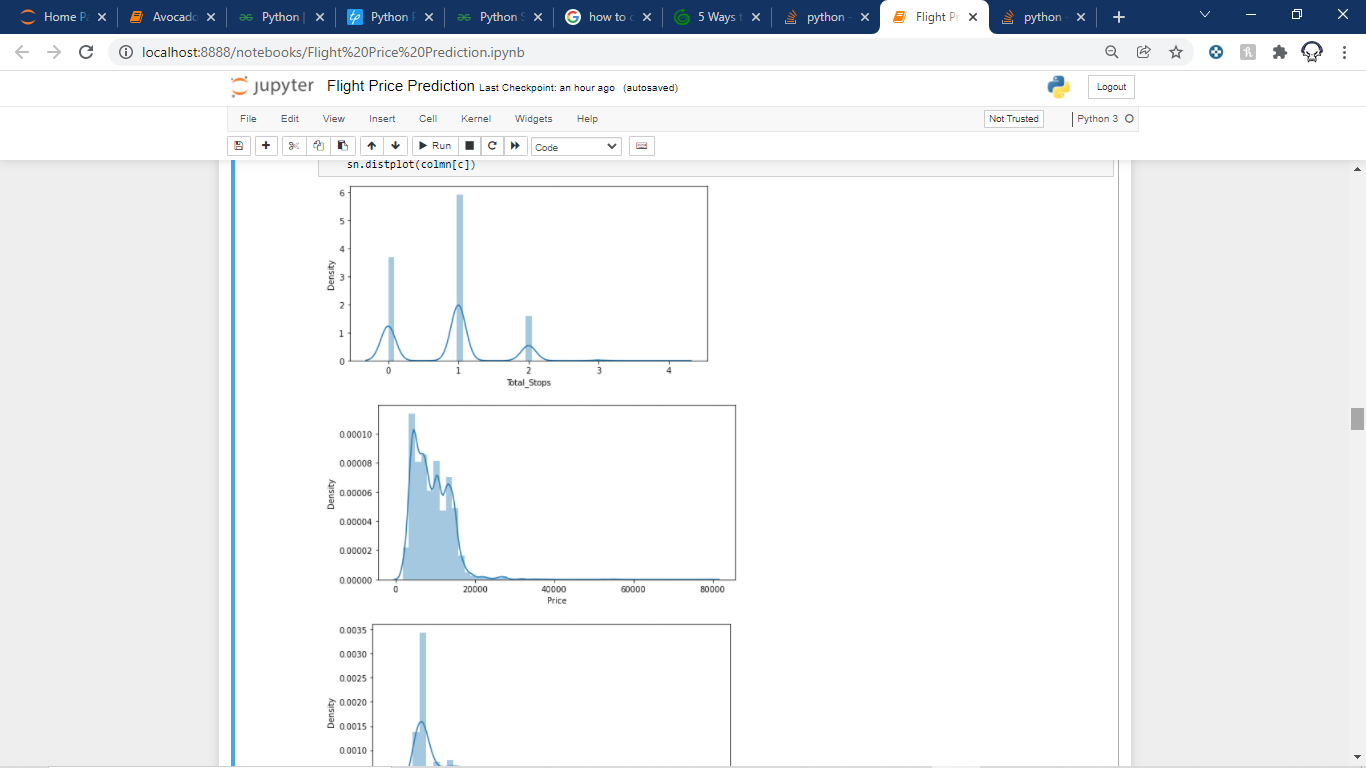
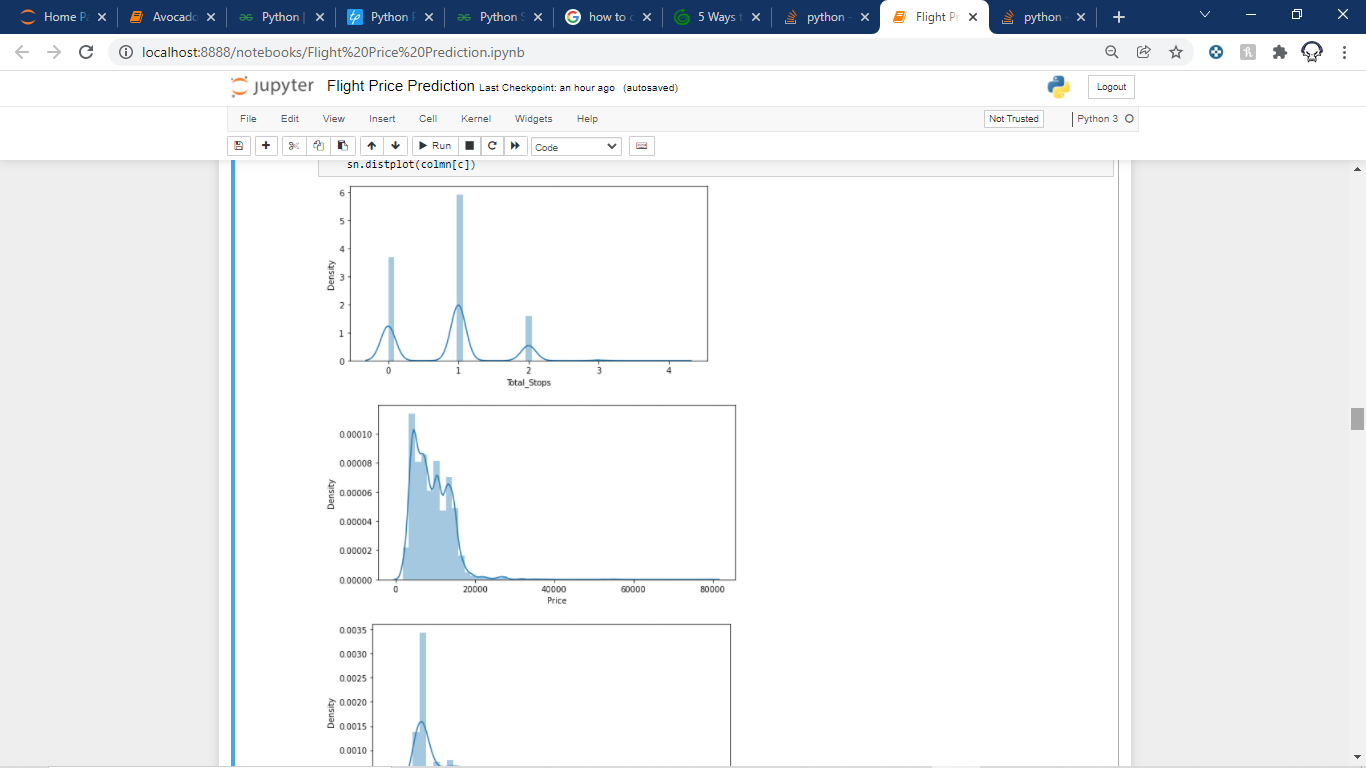
The maximum source is from Bangalore and the least source is from Chennai.

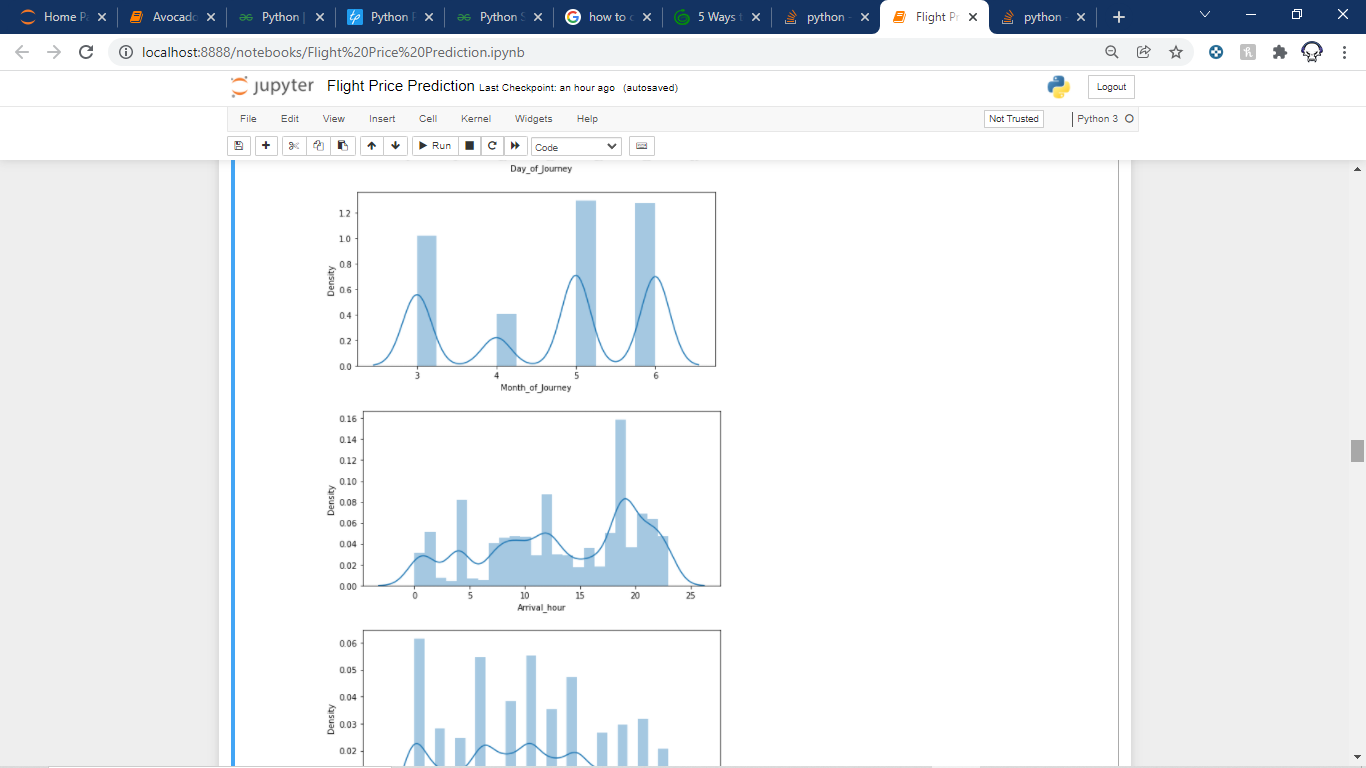
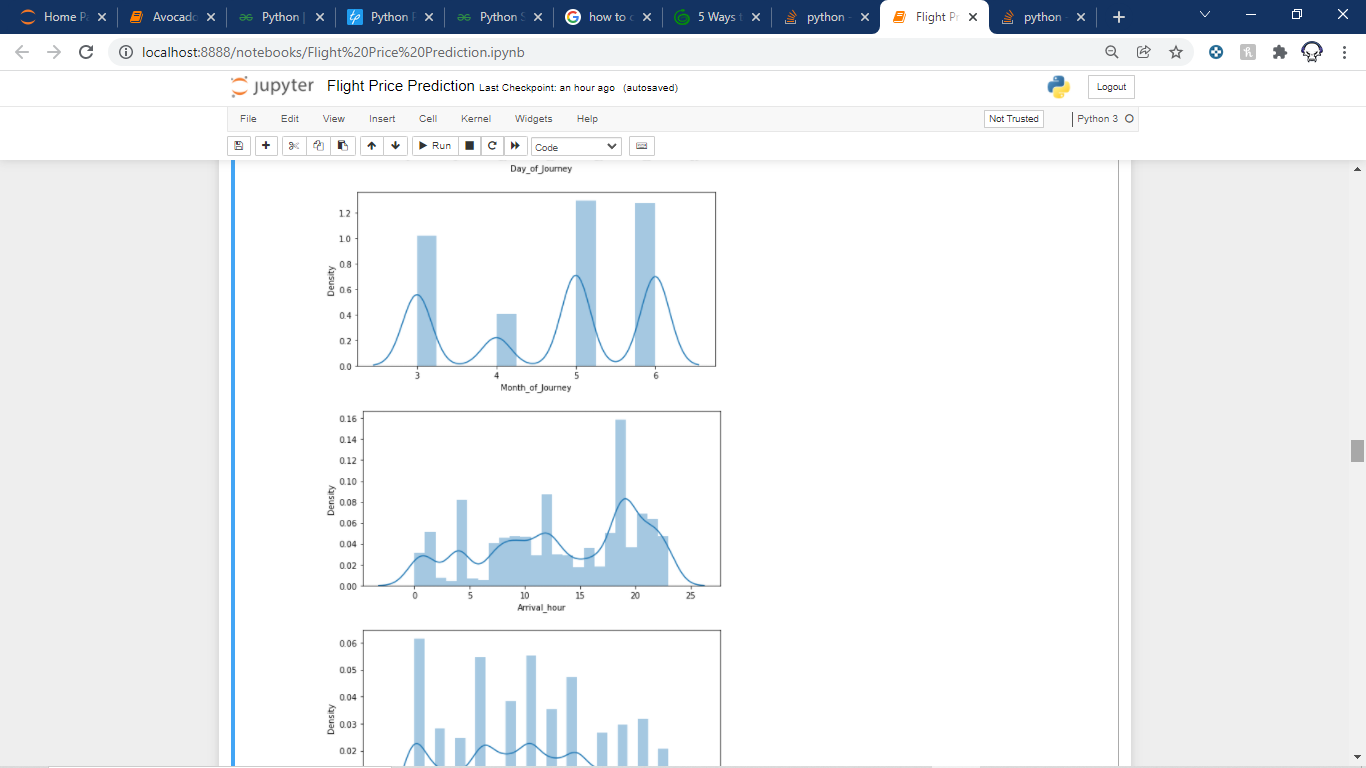
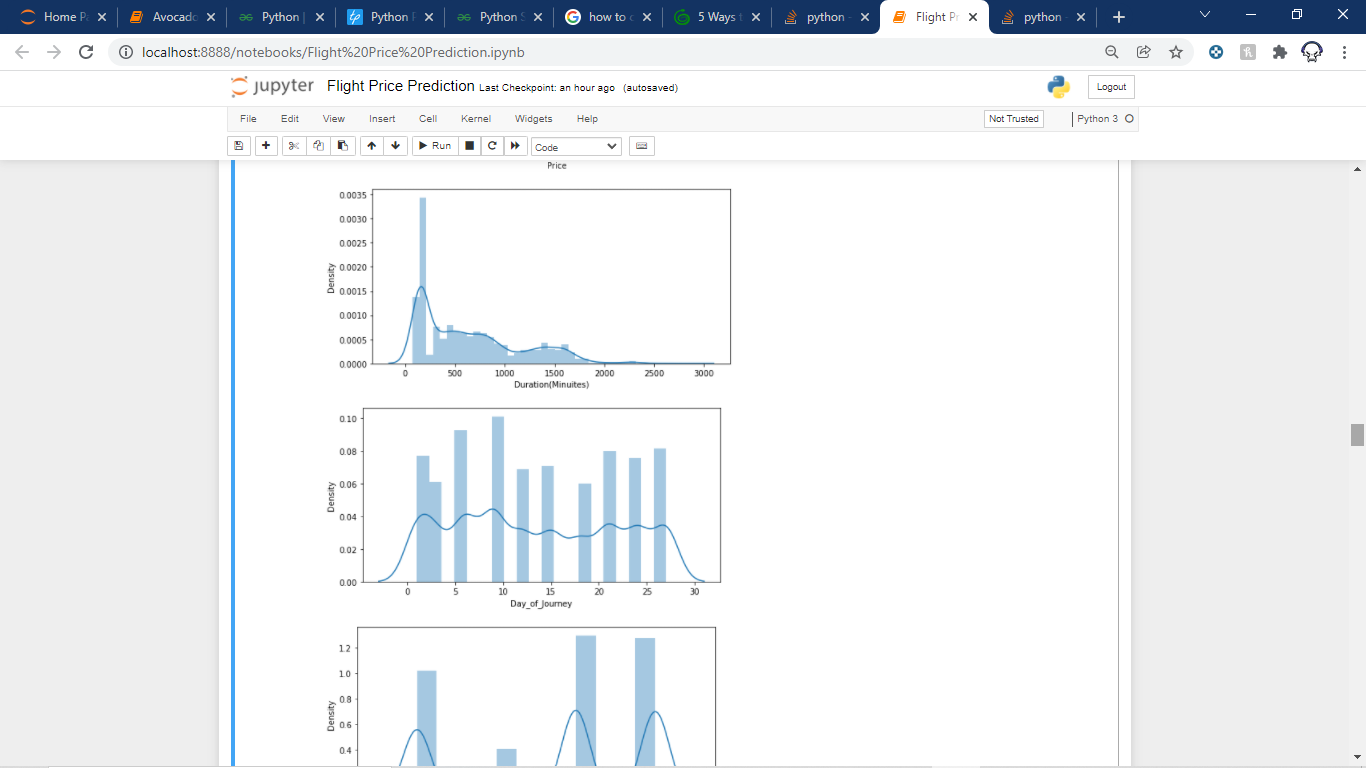
Creating Boxplots:

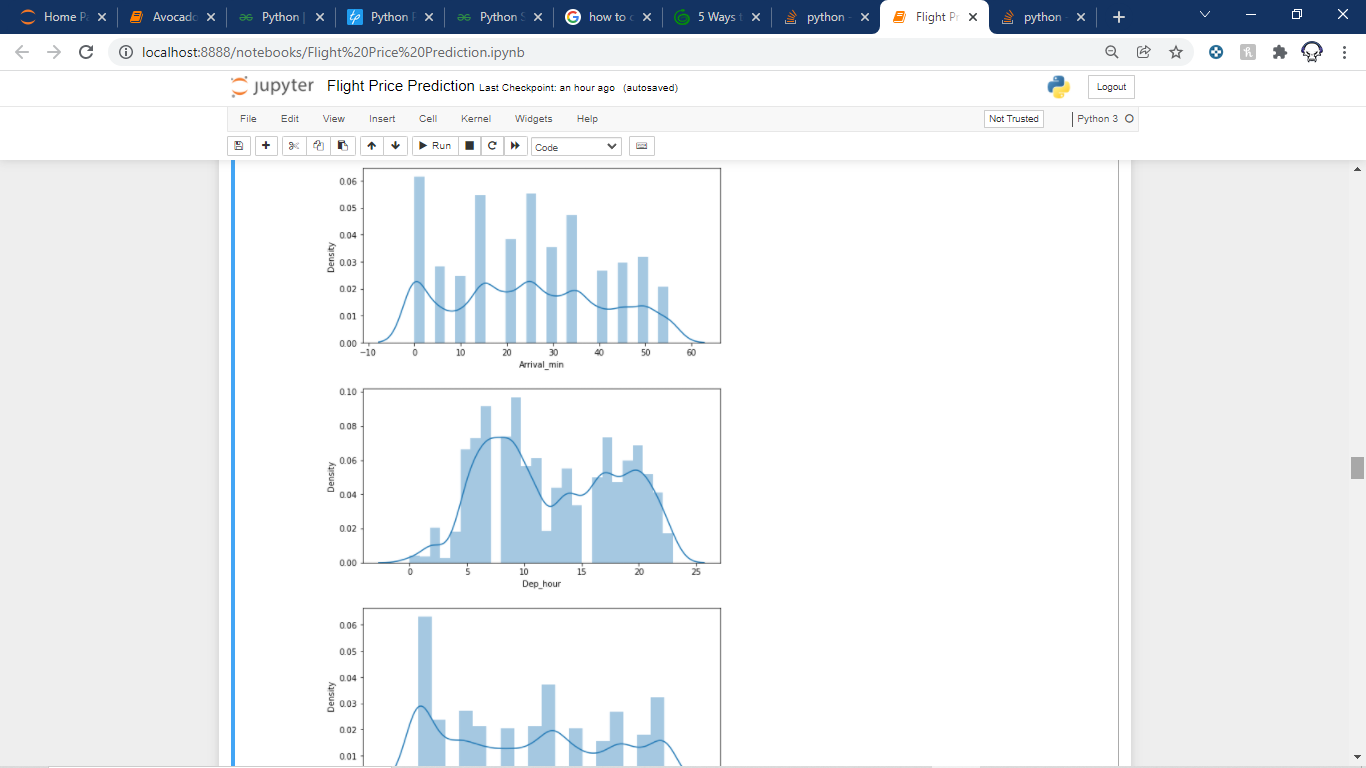
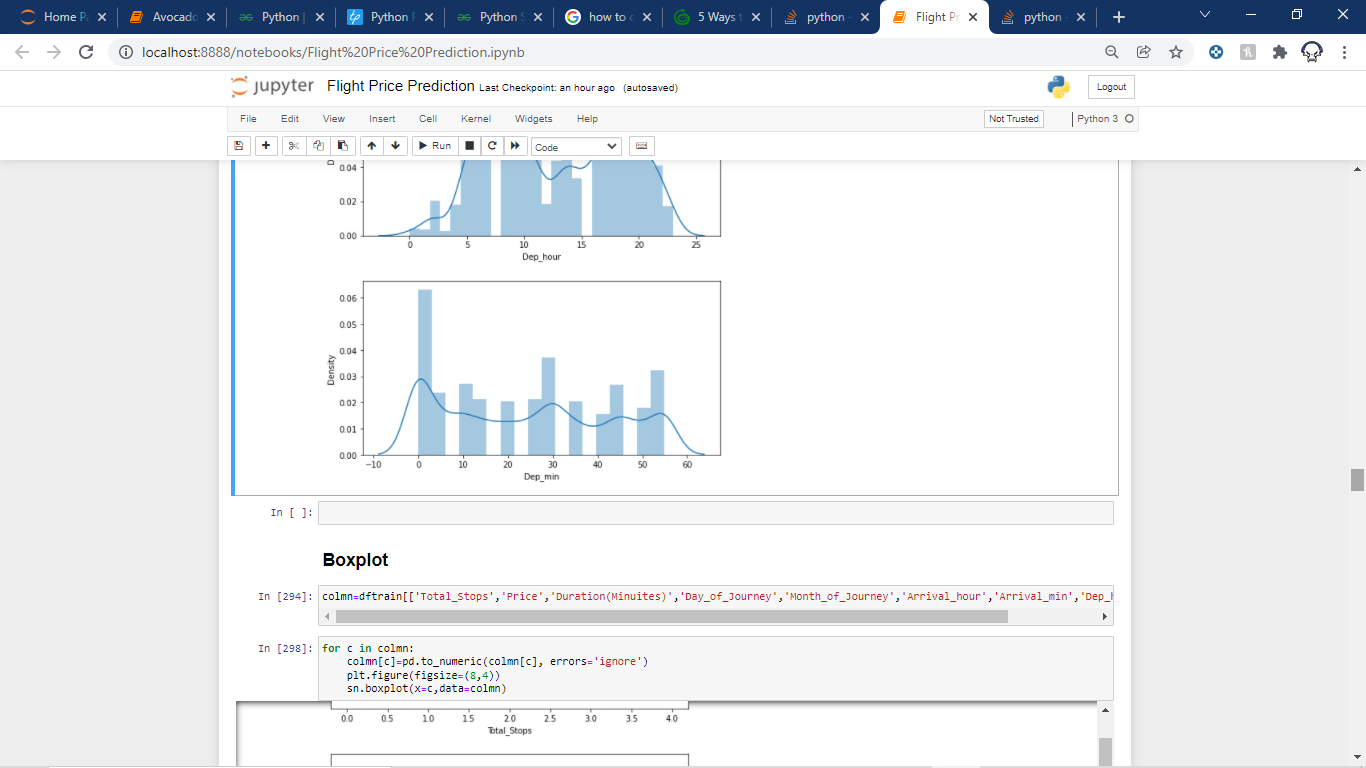
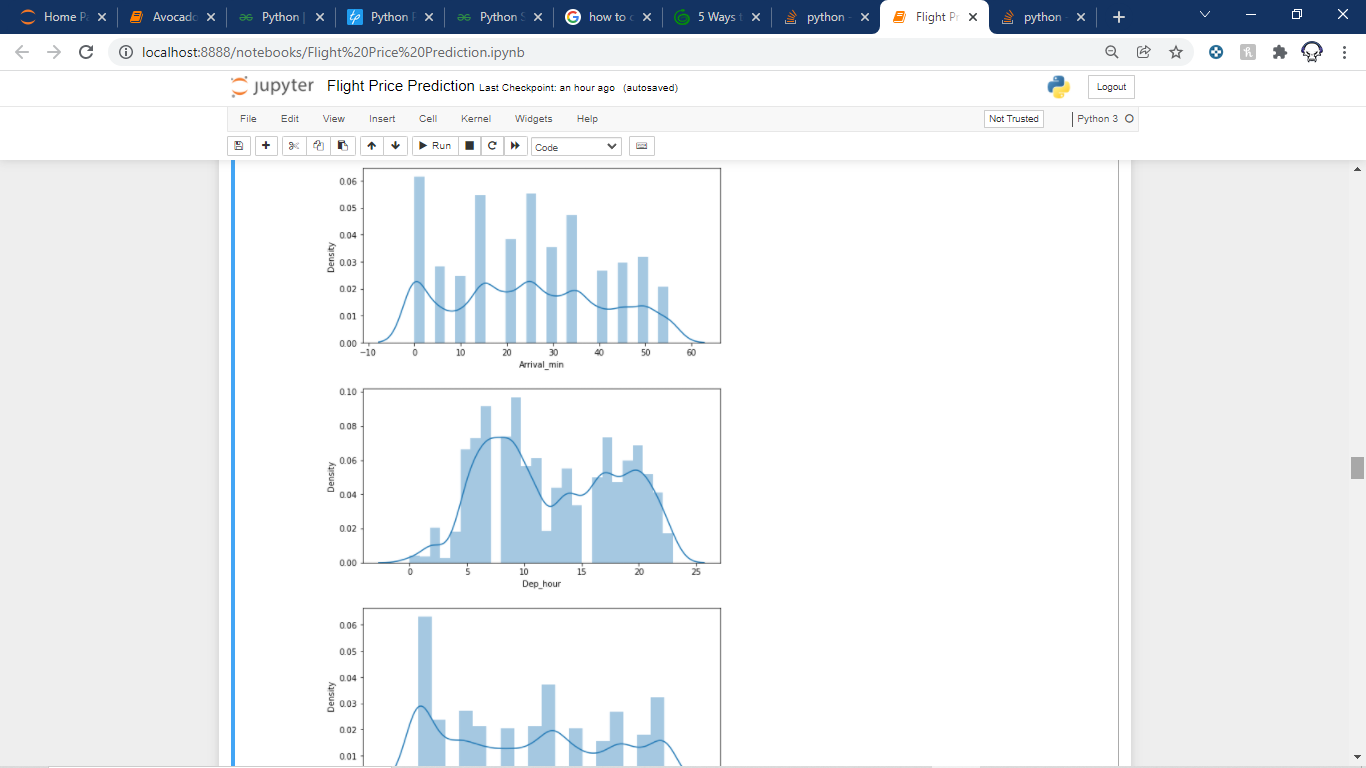


By doing so we realise that the Total\_Stops, Price , Duration(minutes) are the columns that have some outliers and Price, Duration(minutes) has many outliers which are expected.

Distplot:





Observations:

**Total stops**

* Majority of the flights have stops as 1, flights with 3 and 4 stops are quite low

**Price**

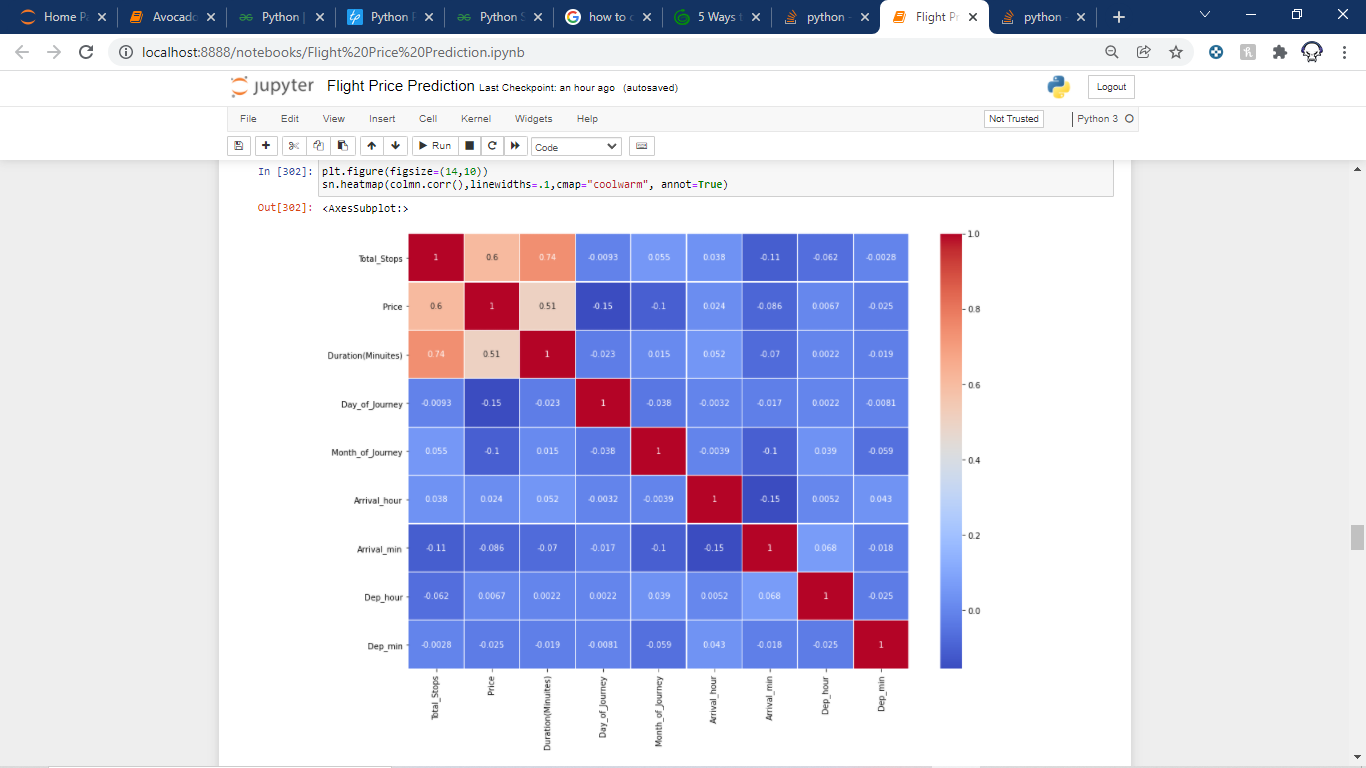
The majority of the flights have a price range between 1759–20k, and the number of flights having prices greater than 20k is quite less. Price is skewed towards the right.

**Duration(Minutes)**

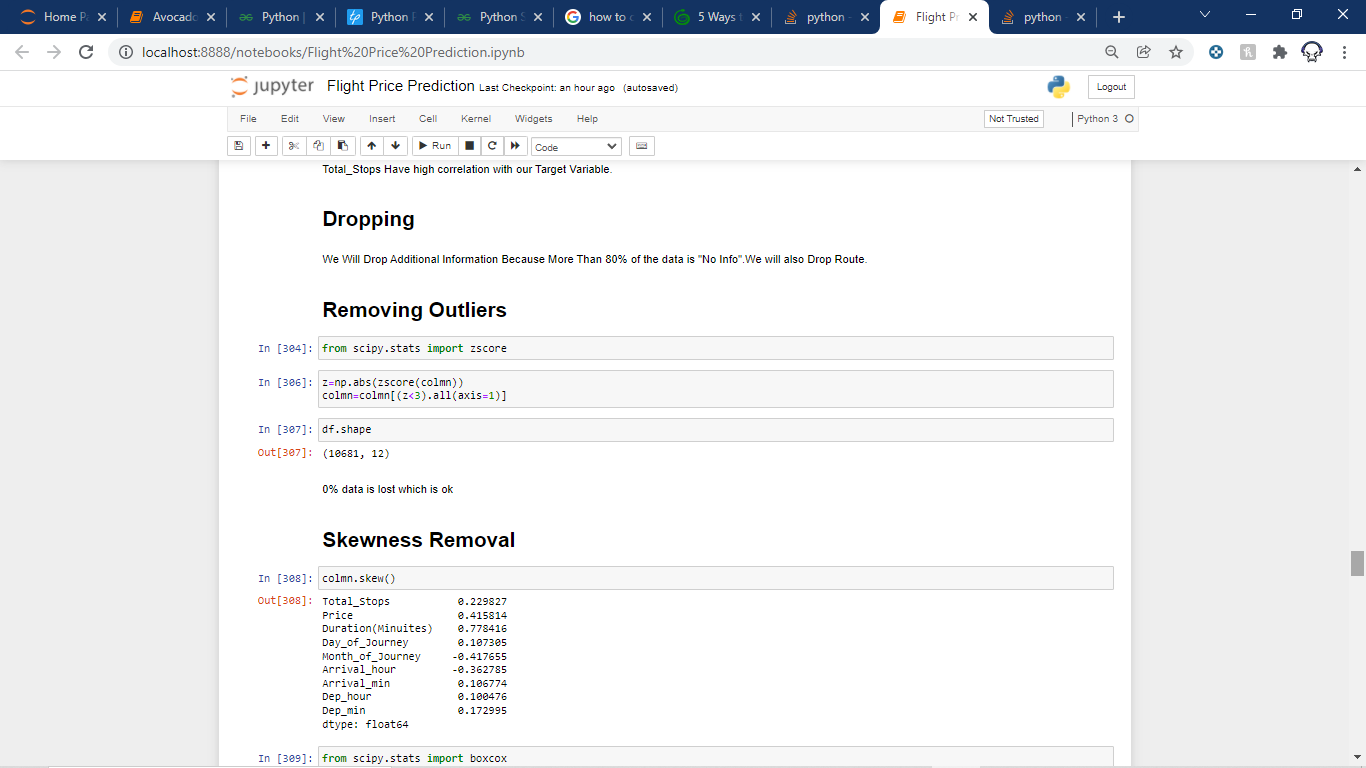
* It is also skewed towards the right.
* The maximum duration is 75 to 80 min flights.Around 2500 flights.
* Majority of the flights have travel time for around 2-3 hours, which seems ok since these are domestic flights
* Some flights have time around 30 hours too, this could be because of the number of stops in between

Let us draw a Heatmap for Certain Columns :

We could see that the columns Total\_stops, Duration(Minutes) have the Highest correlation values among all the other columns. This means as the values of these columns increase the prices also increase.



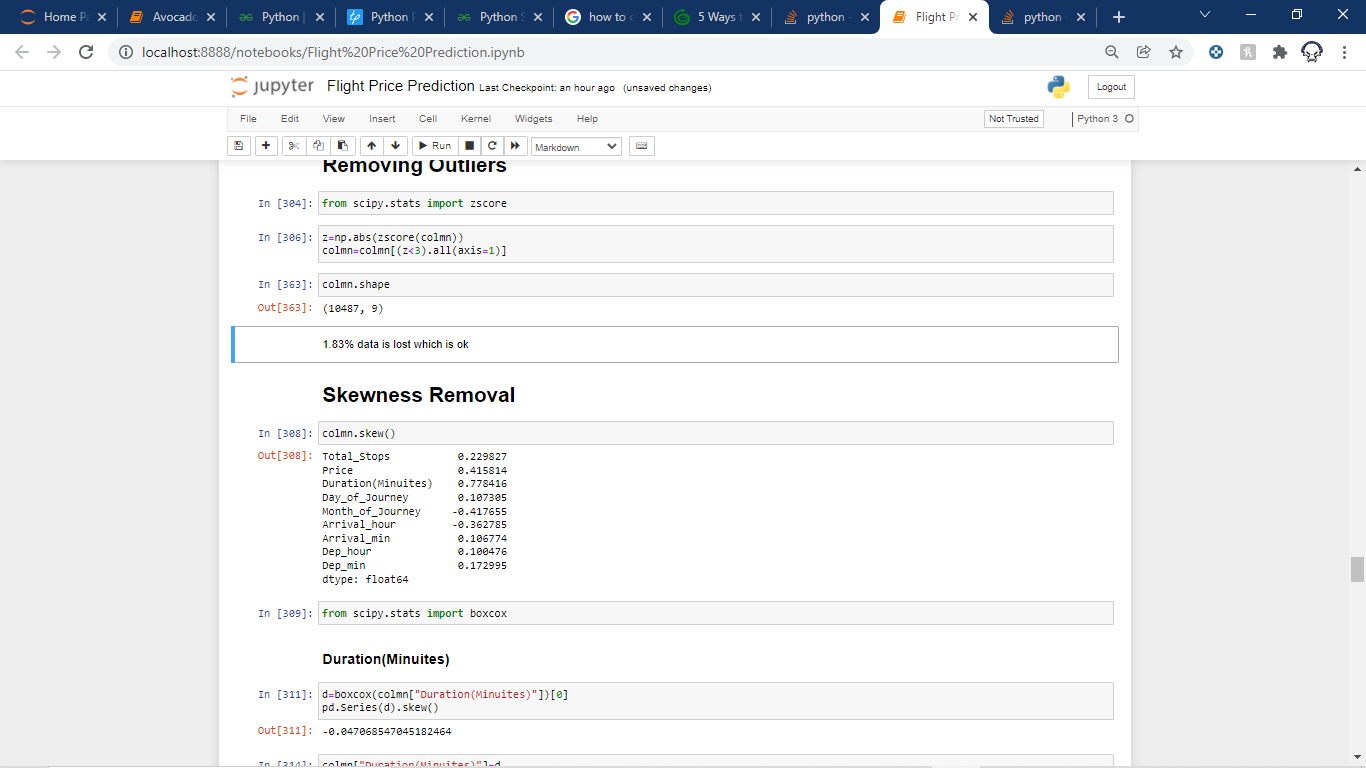
Removing Outliers:



As we can see Only 1.83% of data is lost which is in the acceptable range.

Removing Skewness:

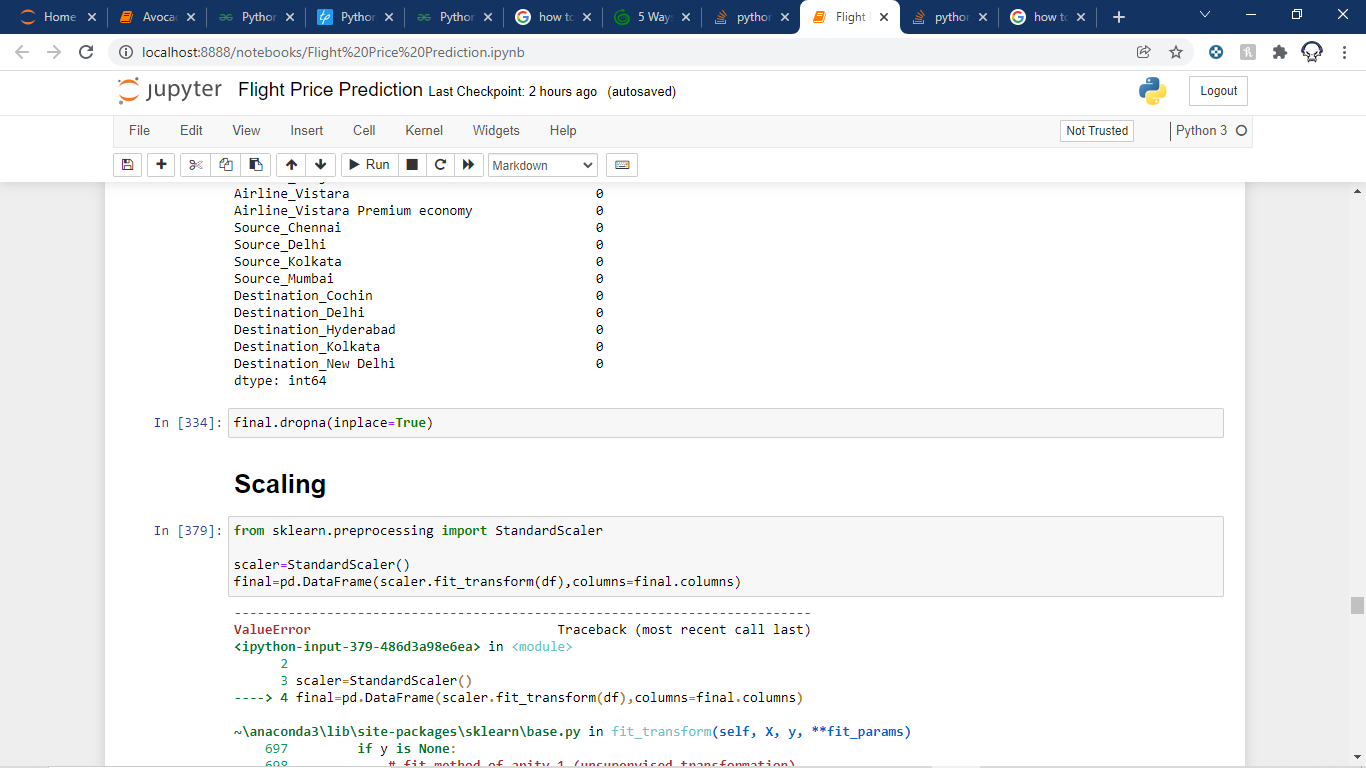
All the other columns have a skewness value of less than 0.5 but Duration(Minutes) has a value greater than that so we have to treat this column and convert it into an acceptable range.



Scaling Our Data:

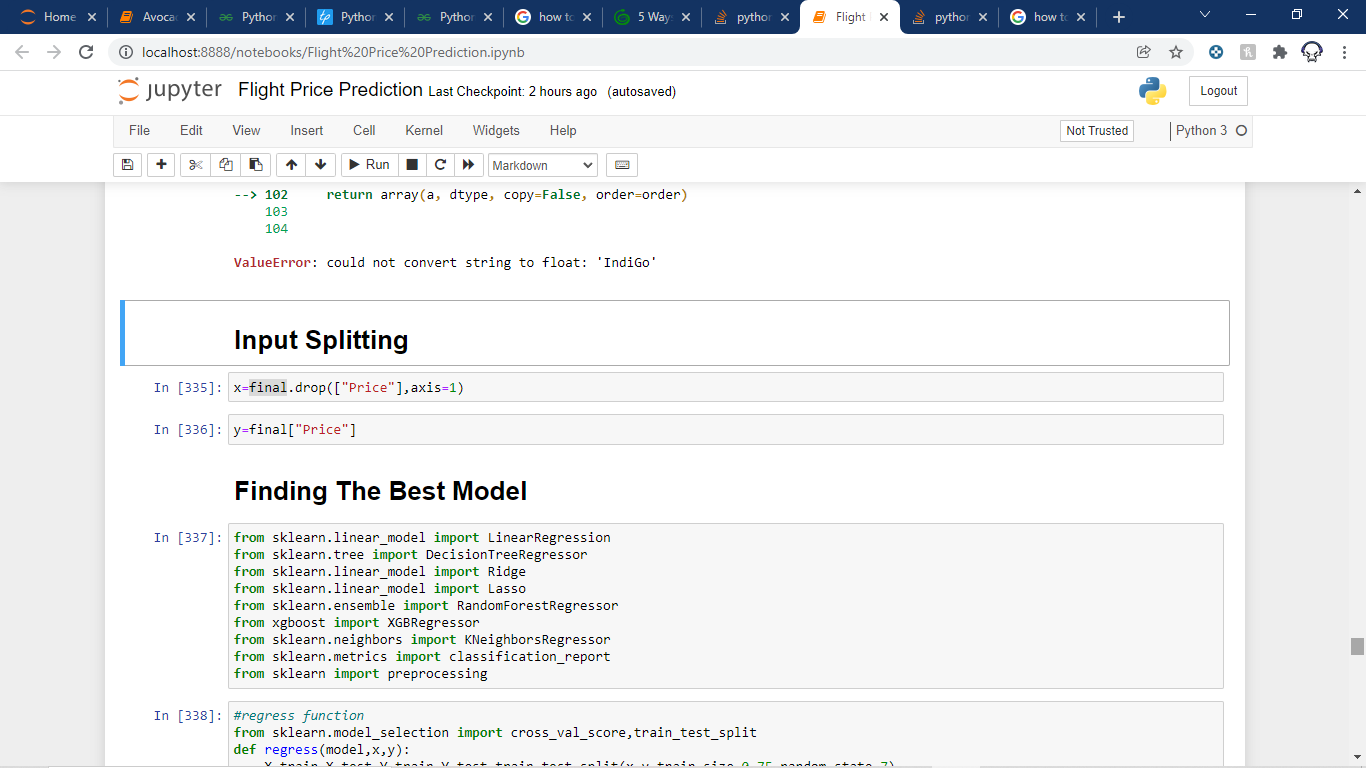
We use standard scaler for this process –

**‘**StandardScaler follows Standard Normal Distribution (SND). Therefore, it makes mean = 0 and scales the data to unit variance’



Input Splitting:

Now we will split the input for Price to feed it into the ML models.

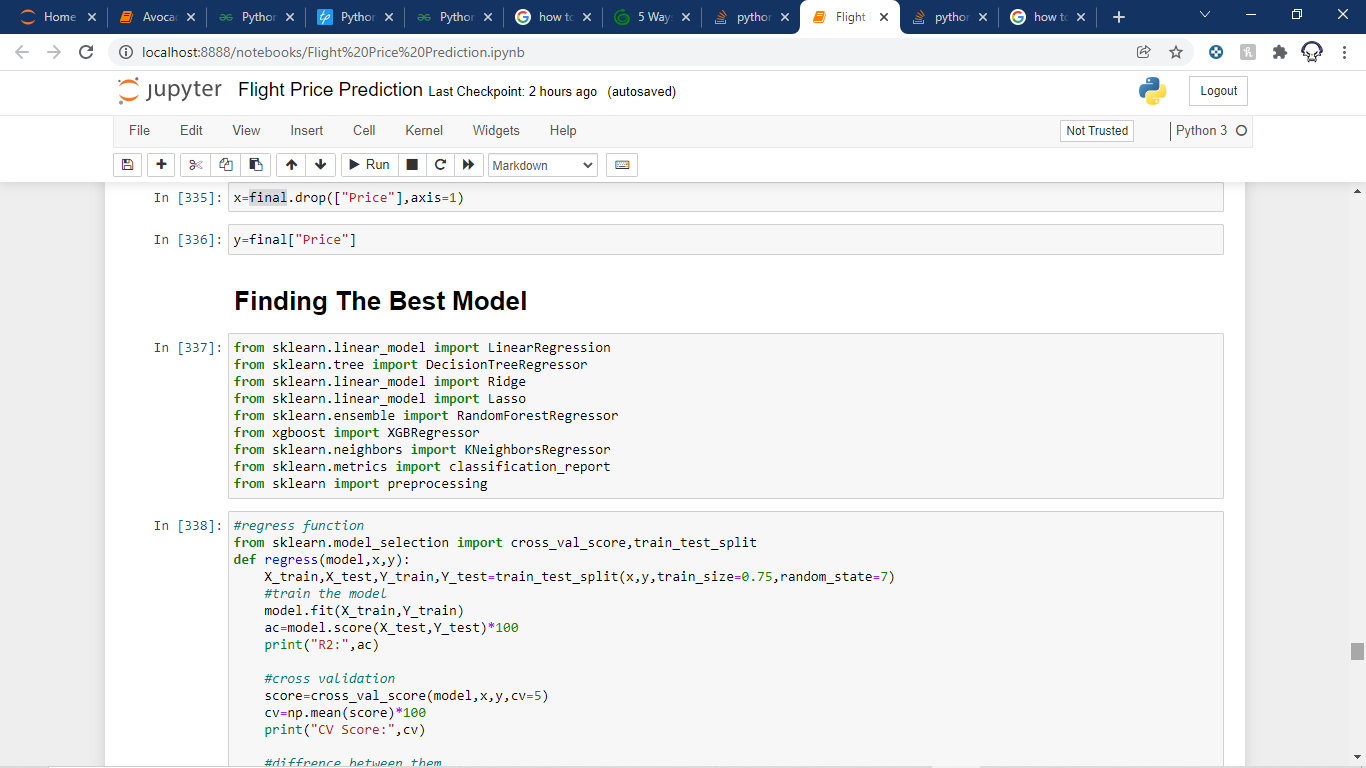


Finding The Best Model:

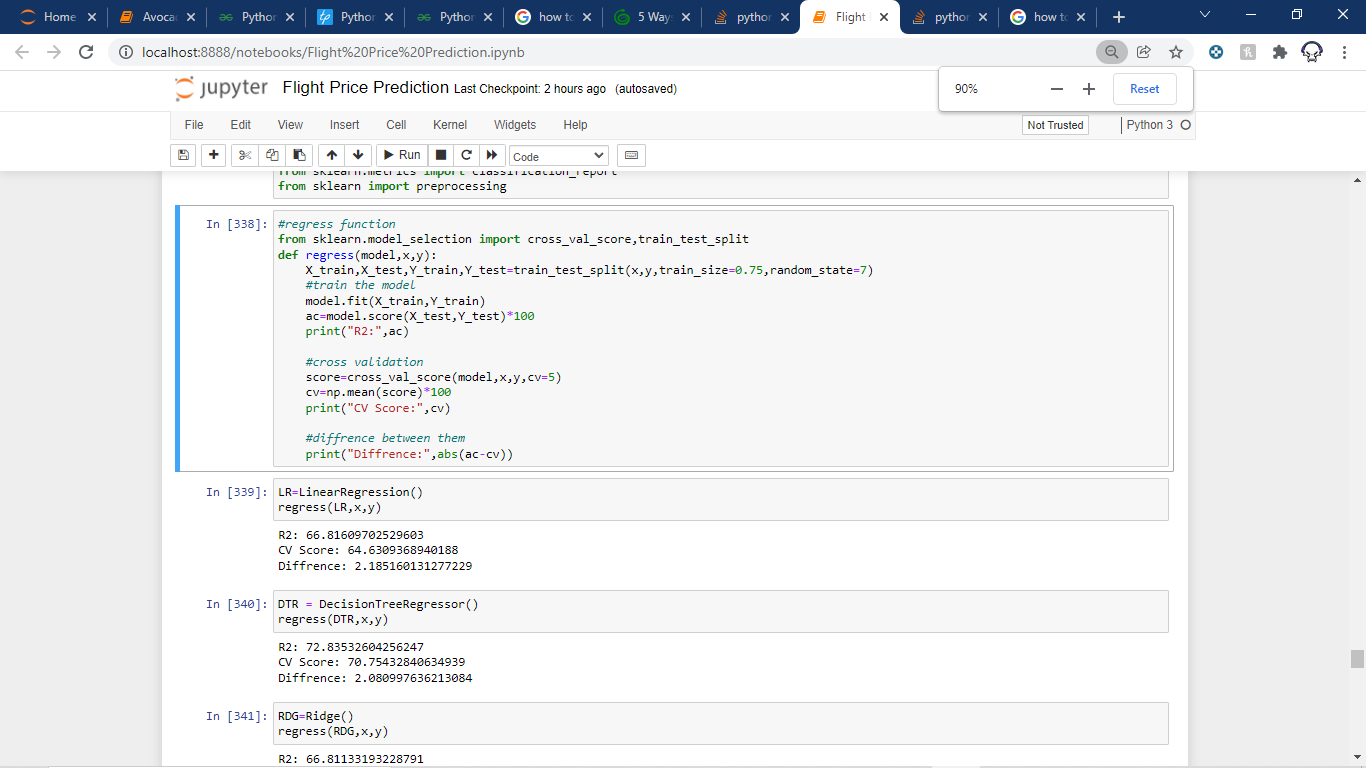
We use various regression models to find out our target variable but only one will be selected.

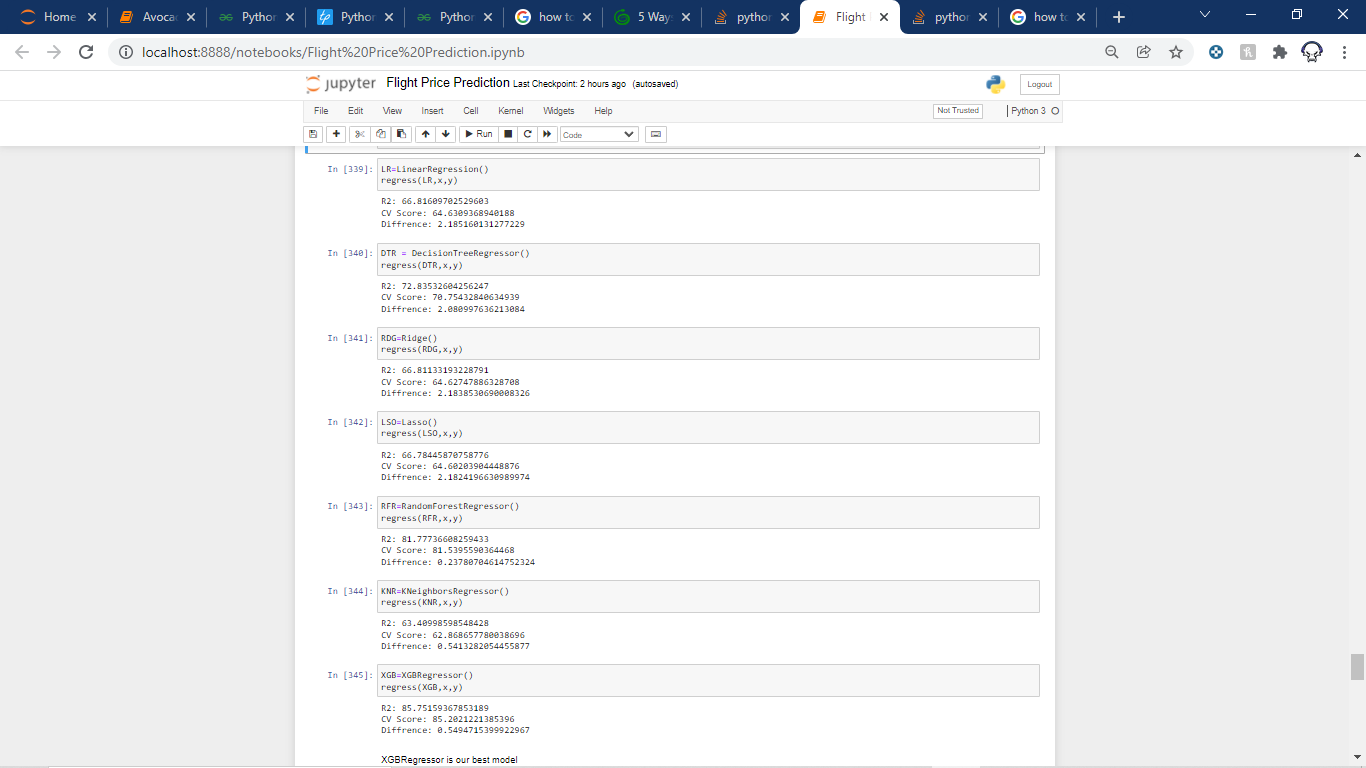
To select the best model out of the bunch we will check the accuracy score of each model but we will not select the model with the highest accuracy score because if we do so then we could face the problem of overfitting. So in order to avoid this problem, we will also calculate the cross-validation score for each model and find out the difference between them. The model with the lowest difference will be our best model and we will proceed further with that model.

Importing The Models:



Finding Out The Best Model:

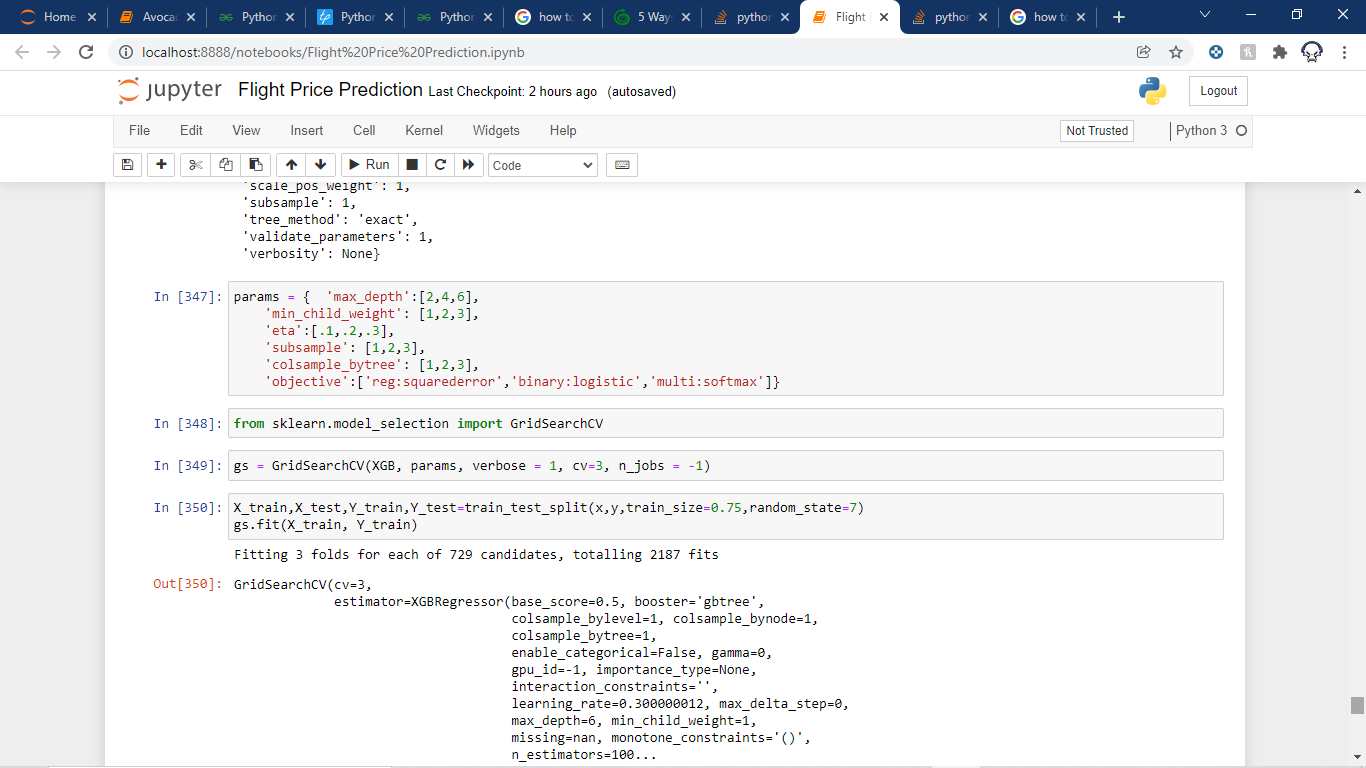




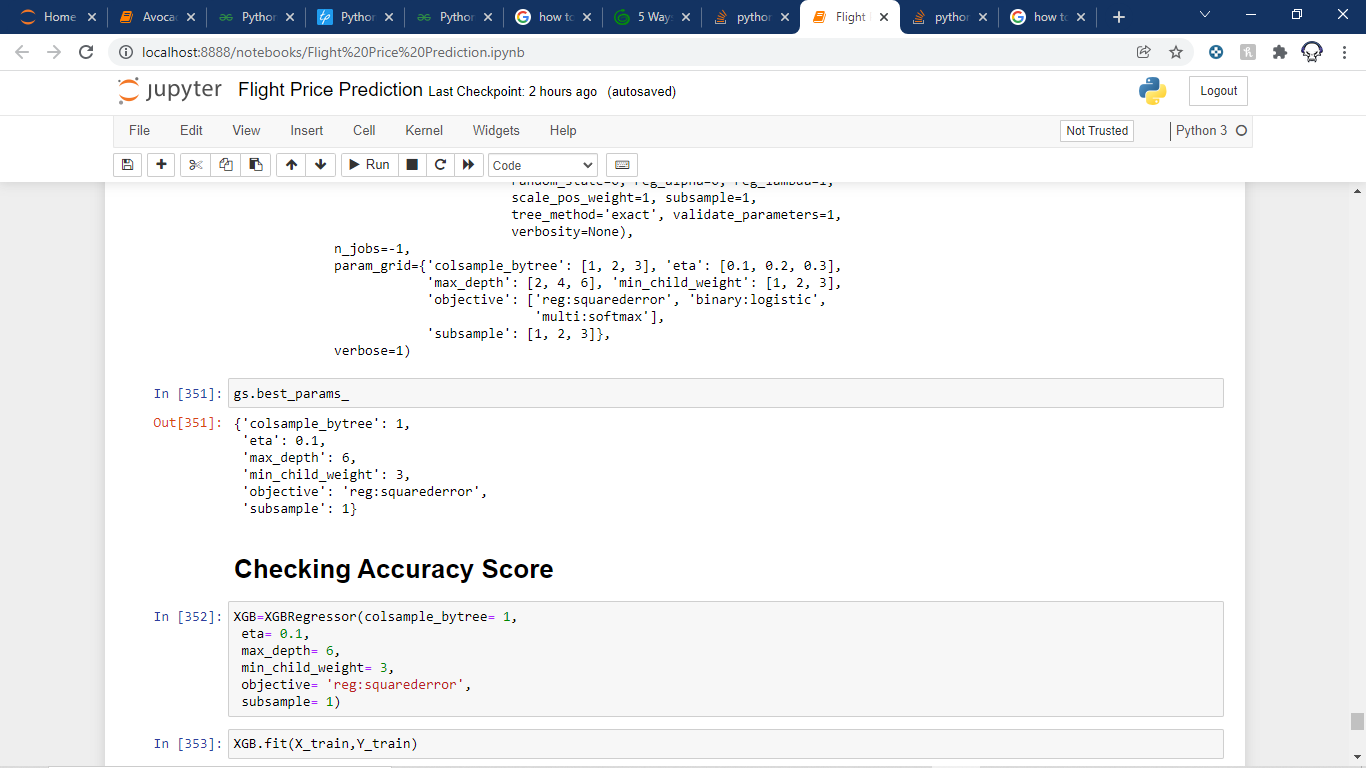
As we can see that XGBoostRegressor is the best model as it has the highest accuracy as well as lowest difference.

Hyper Parameter Tuning:

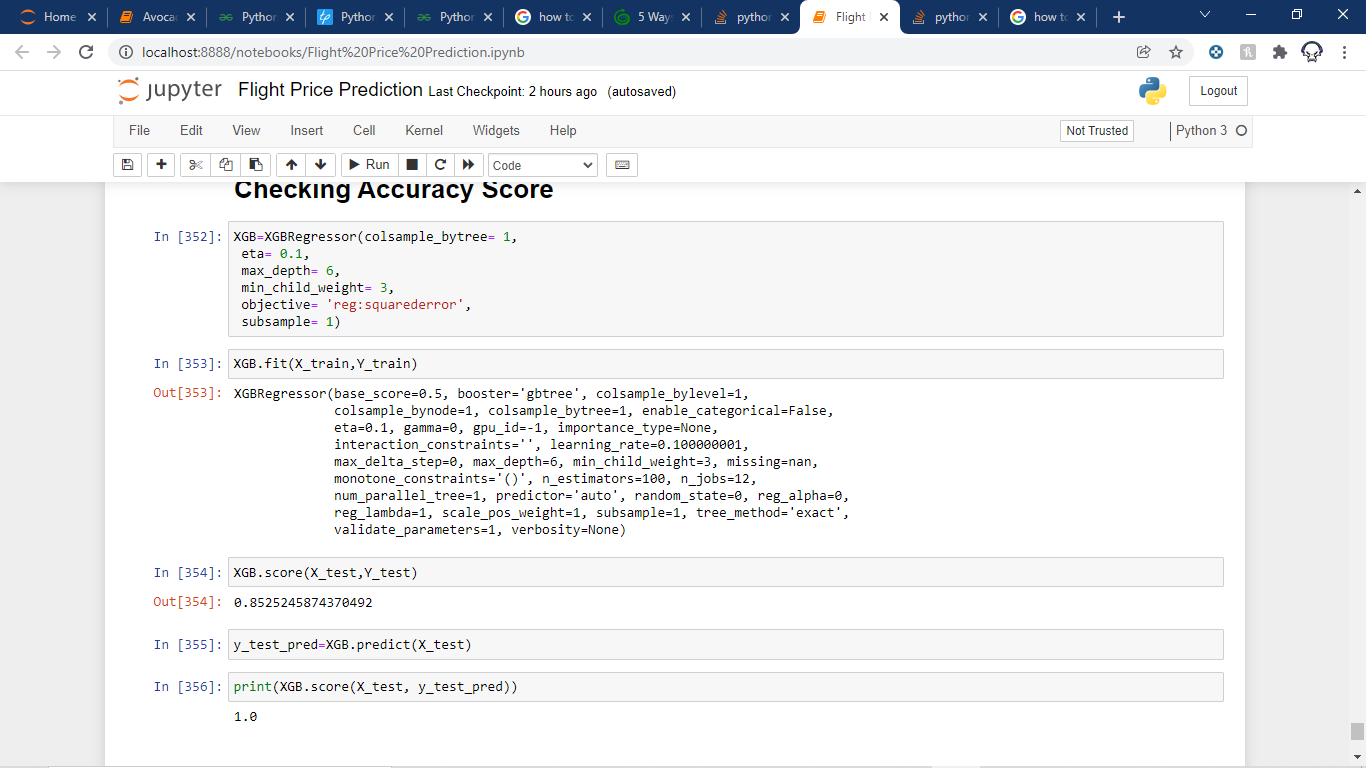
We used GrigSearchCV for this and the following are the parameters we are going to tune:



After fitting we can see the best parameters :

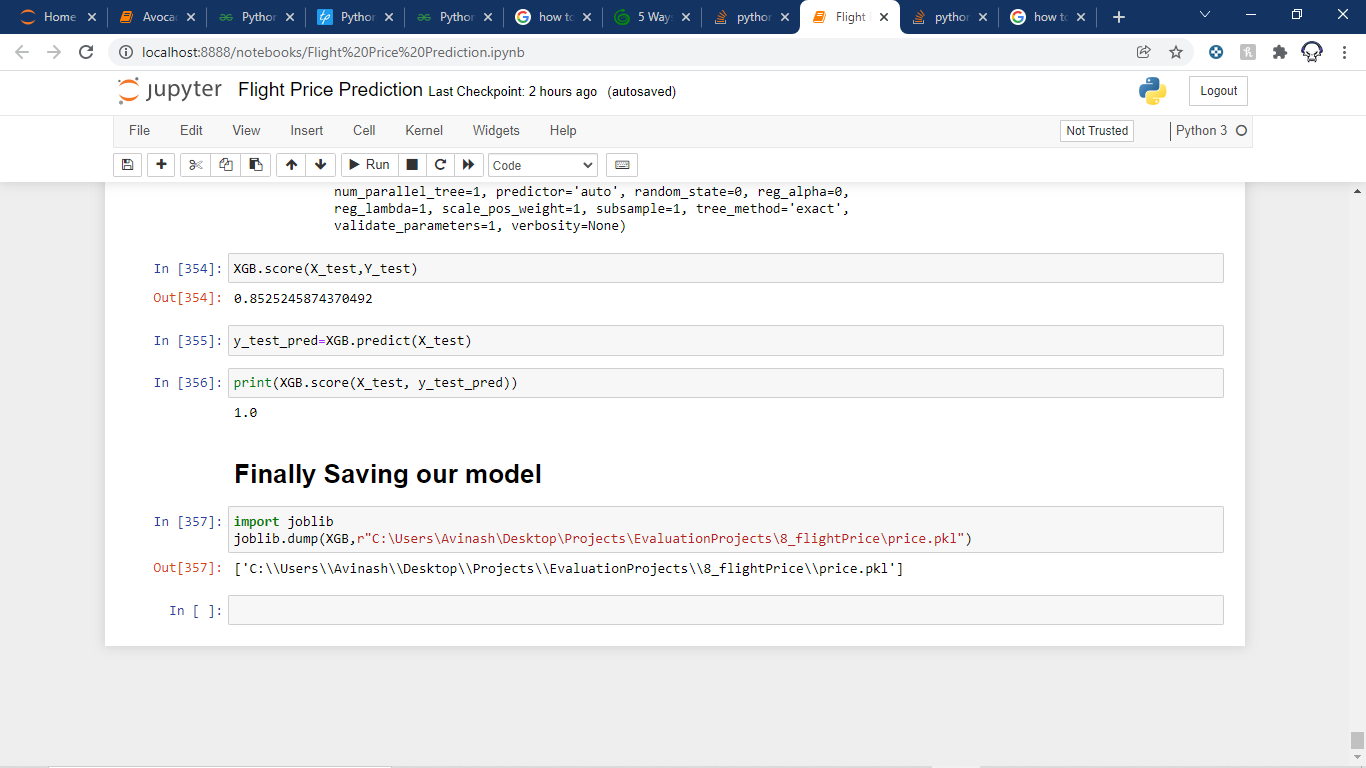


Now Checking the accuracy score last time:



As we can see our Model predicted the test set with 100% accuracy.

Finally Saving Our Model:



Conclusion:

Hence, in the end, we were successfully able to train our regression model “XGBoost” to predict the flights of prices with an r2\_score of 85%, and have achieved the required task successfully.